

Spécifications techniques

[Menu Maker by Qwenta]

Version	Auteur	Date	Approbation
1.0	Ophelie Azor	19/08/2024	John Qwenta

I. Choix technologiques	2
II. Liens avec le back-end.....	3
III. Préconisations concernant le domaine et l'hébergement	3
IV. Accessibilité	3
V. Recommandations en termes de sécurité	3
VI. Maintenance du site et futures mises à jour	4

I. Choix technologiques

- État des lieux des besoins fonctionnels et de leurs solutions techniques :

Besoin	Contraintes	Solution	Description de la solution	Justification (2 arguments)
Création d'une catégorie de menu	L'ajout d'une catégorie doit pouvoir se faire directement sur l'écran de création de menu depuis une modale.	react-modal	Cette librairie React permet de créer simplement des modales performantes, accessibles avec un minimum de code.	1) Nous avons choisi de développer en React, la librairie est cohérente avec ce choix. 2) Il s'agit de la librairie la plus utilisée.
Mentions légales et "Tous droits réservés »	Le contenu des mentions légales doit être accessible via une modale sur toutes les pages, avec "Tous droits réservés" visible partout.	HTML/CSS/JavaScript	Utiliser HTML pour structurer les mentions légales et la mention "Tous droits réservés", CSS pour le style de la modale, et JavaScript pour gérer l'affichage de la modale.	1) HTML/CSS assurent une présentation claire et uniforme du contenu statique des mentions légales. 2) JavaScript permet une interaction fluide avec la modale pour l'affichage des mentions.

Authentification	Authentification sécurisée, gestion de session	MongoDB avec Mongoose et bcrypt	Utilisation de MongoDB avec des bibliothèques comme Mongoose pour gérer l'inscription, la connexion et la gestion des utilisateurs. bcrypt est utilisé pour le hashage des mots de passe.	1) Sécurité : MongoDB, combiné avec bcrypt, offre une solution sécurisée pour le stockage des mots de passe en utilisant des techniques de hashage robustes. 2) Flexibilité : MongoDB permet de stocker les données des utilisateurs dans une structure flexible, simplifiant l'évolution des modèles d'authentification et de gestion des sessions.
Dashboard	Le dashboard doit regrouper des encarts pour créer, diffuser, et imprimer des menus, ainsi que des articles de blog.	HTML/CSS/JavaScript	Utiliser HTML pour structurer les encarts du dashboard, CSS pour le style, et JavaScript pour la gestion dynamique des articles et des liens.	1) HTML/CSS fournissent la structure et le style nécessaires pour présenter les encarts et les articles de manière claire. 2) JavaScript permet d'intégrer et de mettre à jour dynamiquement les articles du blog et les liens, offrant une expérience utilisateur fluide.
Landing Page Non Connectée	La page doit inclure une bannière, une section pour personnaliser le menu, et des explications étape par étape.	React/HTML/CSS	Utiliser React pour créer des composants dynamiques (bannière, sections interactives) et CSS pour le style et la mise en page.	1) React permet des éléments interactifs et dynamiques, améliorant l'engagement utilisateur. 2) CSS assure un design cohérent et attrayant.

Exportation PDF	Le restaurateur doit pouvoir télécharger le fichier PDF en un clic.	HTML/CSS/JavaScript /PDF Library	Utiliser une bibliothèque JavaScript comme jsPDF pour générer le fichier PDF à partir des données du menu, et intégrer un bouton pour l'exportation.	1) jsPDF est largement utilisé pour la génération de fichiers PDF côté client, offrant une solution efficace pour les besoins de création de PDF. 2) HTML/CSS assurent une présentation cohérente du menu avant l'exportation.
Partage sur Instagram	Le restaurateur doit pouvoir partager son menu en format carré sur Instagram et être redirigé vers son compte.	HTML/CSS/JavaScript /Instagram API	Utiliser HTML et CSS pour générer des images du menu au format carré, JavaScript pour gérer le téléchargement des images, et intégrer l'API Instagram pour redirection.	1) HTML/CSS sont utilisés pour créer des images bien formatées du menu, assurant une bonne présentation sur Instagram. 2) JavaScript permet la gestion du téléchargement et de la redirection, simplifiant le partage du contenu
Imprimer le menu	L'encart "Imprimer un menu" doit être visible sur la page d'accueil et rediriger vers le back-office de Qwenta.	HTML/CSS/JavaScript	Ajouter un encart "Imprimer un menu" sur la page d'accueil qui, lorsqu'on clique dessus, ouvre un lien vers le back-office de Qwenta dans un nouvel onglet.	1) HTML/CSS permettent d'ajouter et styliser l'encart de manière visible et intuitive sur la page d'accueil. 2) JavaScript facilite l'ouverture du lien dans un nouvel onglet, assurant une transition fluide vers le back-office.

II. Liens avec le back-end

- Quel langage pour le serveur ? Nous allons utiliser NodeJS avec Express.
- A-t-on besoin d'une API ? Si oui laquelle ? API REST, API Instagram, API Deliveroo
- Base de données choisie : Une base de données NOSQL nommé MongoDB

III. Préconisations concernant le domaine et l'hébergement

- Nom du domaine : **Utiliser un sous-domaine dédié sous le domaine principal de Qwenta.**
- Nom de l'hébergement : **Google Cloud Platform,**
- Adresses e-mail : info@qwenta.com, support@qwenta.com

IV. Accessibilité

- Compatibilité navigateur

Le site doit être compatible avec les dernières version de Chrome, Firefox et Safari.

- Types d'appareils

On pourra visualiser le site avec un ordinateur, une version mobile n'est pas prévue pour le moment.

V. Recommandations en termes de sécurité

Accès aux comptes et gestion des mots de passe :

1. JWT (Jsonwebtoken) :

- Utiliser JWT (jsonwebtoken) pour l'authentification des utilisateurs. Les JWT doivent être signés et si possible, chiffrés pour garantir leur intégrité et confidentialité.
- Configurer une durée d'expiration pour les tokens. Les tokens doivent avoir une durée de vie limitée pour réduire les risques si un token est compromis

2. HTTPS :

- Utiliser https pour toutes les communications entre le client et le serveur afin de chiffrer les données en transit et protéger contre les attaques.
- S'assurer que le serveur utilise des certificats SSL valides et à jour pour établir une connexion sécurisée

3. Journalisation des Accès :

- Enregistrer les tentatives de connexion, les changements de mot de passe, et les activités associées aux tokens dans des logs de sécurité. Analyser ces logs pour détecter des comportements suspects..

4. Stockage sécurisé des mots de passe :

- Utiliser un algorithme de hachage sécurisés comme bcrypt pour stocker les mots de passe. Ces algorithmes rendent le mot de passe illisible et difficile à déchiffrer.

5. Alerte de connexion :

- Envoyer des notifications par e-mail ou par SMS pour informer les utilisateurs de nouvelles connexions ou tentatives de connexion depuis des emplacements inconnus.

6. Conseils de sécurité :

- Fournir des recommandations et des meilleures pratiques de sécurité aux utilisateurs pour les aider à créer des mots de passe sécurisés et à reconnaître les tentatives de phishing.

VI. Maintenance du site et futures mises à jour

Certaines fonctionnalités pourront être ajoutées par la suite comme ci-dessous :

- Grandes lignes du contrat de maintenance : Le contrat de maintenance inclura des mises à jour régulières, des correctifs de sécurité et des optimisations de performance pour garantir la stabilité et la sécurité du site.
- Ajouter des animations sur la photo de la bannière et sur les formes géométriques des sections.
- Intégrer le tarif directement sur Menu Maker
- Créer la possibilité de changer de moyen de paiement
- Possibilité de créer un blog interne à Menu Maker

De plus, des tests réguliers seront effectués pour identifier et corriger les bugs potentiels, assurer une expérience utilisateur fluide, et garantir la compatibilité avec les nouvelles technologies. Les mises à jour seront planifiées de manière préventive pour éviter les interruptions de service et maintenir un haut niveau de performance et de sécurité.